PlanetServer tutorial

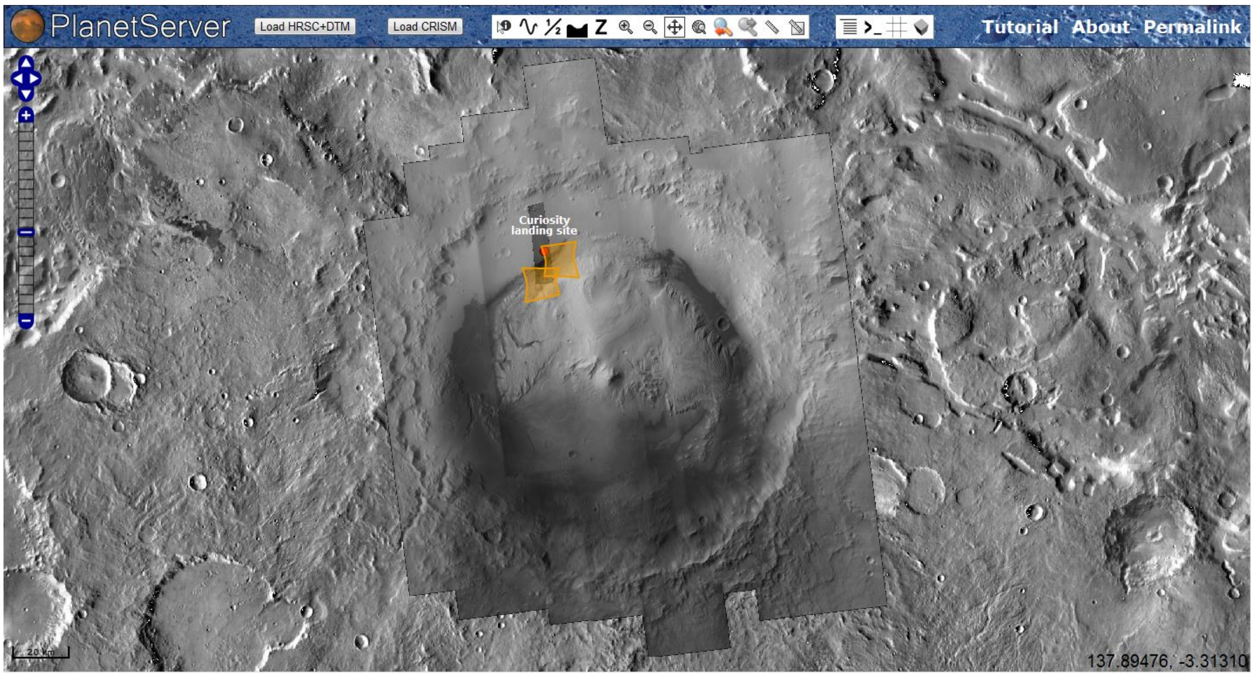
Contents

Introduction

PlanetServer (www.planetserver.eu) is an online analysis tool for planetary data (currently focusing on the planet Mars). It is part of the EarthServer project (www.earthserver.eu).

PlanetServer uses the Rasdaman Array Database (www.rasdaman.org) on the server side and OpenLayers (www.openlayers.org) on the client side. Communication between the client and server takes place using Open Geospatial Consortium (OGC, www.opengeospatial.org) protocols:

1. Web Map Service (WMS) to visualize visual imagery.
2. Web Coverage Processing Service (WCPS) to allow access to elevation and hyperspectral data.



Two demonstrations have been created:

1. **HRSC+DTM analysis**, demonstrating 3D visualization through X3D (www.x3dom.org) and extracting elevation cross sections and elevation point values through WCPS. HRSC stands for High Resolution Stereo Camera, a camera on board the ESA Mars Express satellite. It consists of multiple imagers which are positioned in different angles allowing the creation of stereo derived elevation models.
2. **CRISM data analysis**, demonstrating spatial and spectral analysis of hyperspectral data using WCPS. CRISM stands for Compact Reconnaissance Imaging Spectrometer for Mars, a hyperspectral imager on board the NASA Mars Reconnaissance Orbiter satellite.

Recommendations

1. PlanetServer (www.planetserver.eu) has been tested on Mozilla FireFox and Google Chrome. If another browser is used the user will be automatically forwarded to: http://planetserver.jacobs-university.de/browser.html
2. The X3D demo needs WebGL enabled.
    a. FireFox: about:config -> enable WebGL
    b. Chrome: chrome.exe --disable-web-security
3. Please use a large screen

Issues

1. Google Chrome doesn't show the Loader image properly. The Loader image is shown every time when in the background data is loaded from the server.
2. The X3D feature has been successfully tested on Google Chrome and Firefox. The OS type and graphics hardware can also have an influence. If it doesn't work on Google Chrome, please try FireFox, and vice versa.

HRSC + DTM analysis

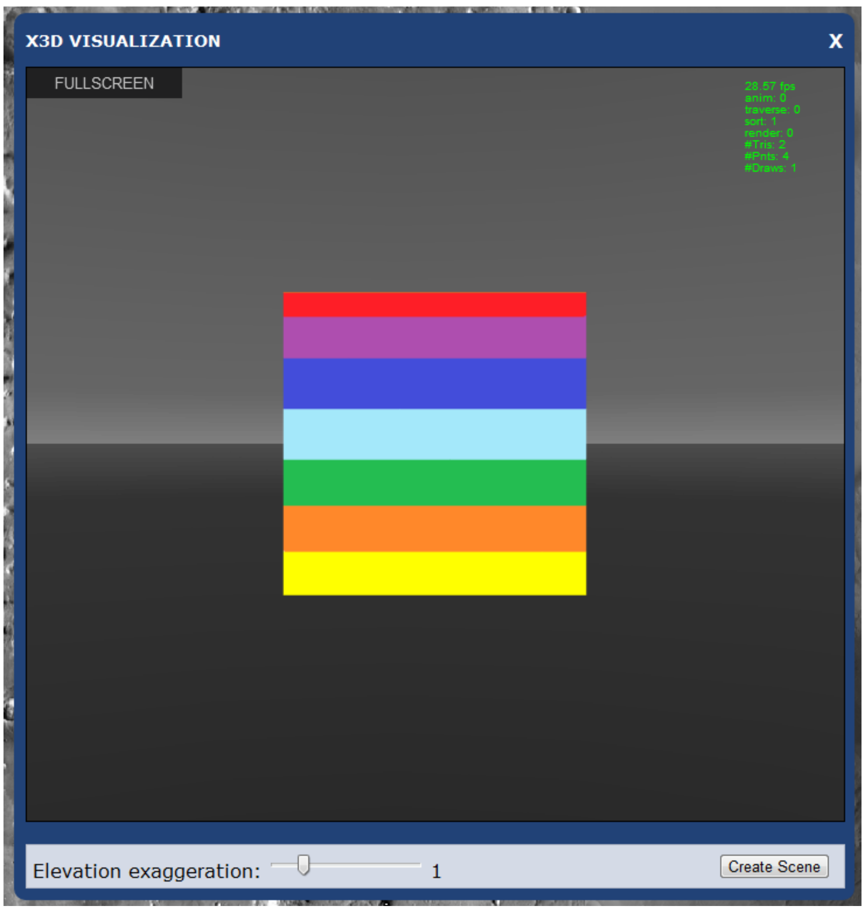
Clicking on Load HRSC+DTM starts the HRSC + DTM analysis demonstration. Two windows appear:

1. A X3D window appears on the left side of the screen
2. The Cross Section tab of the diagram window appears on the right.

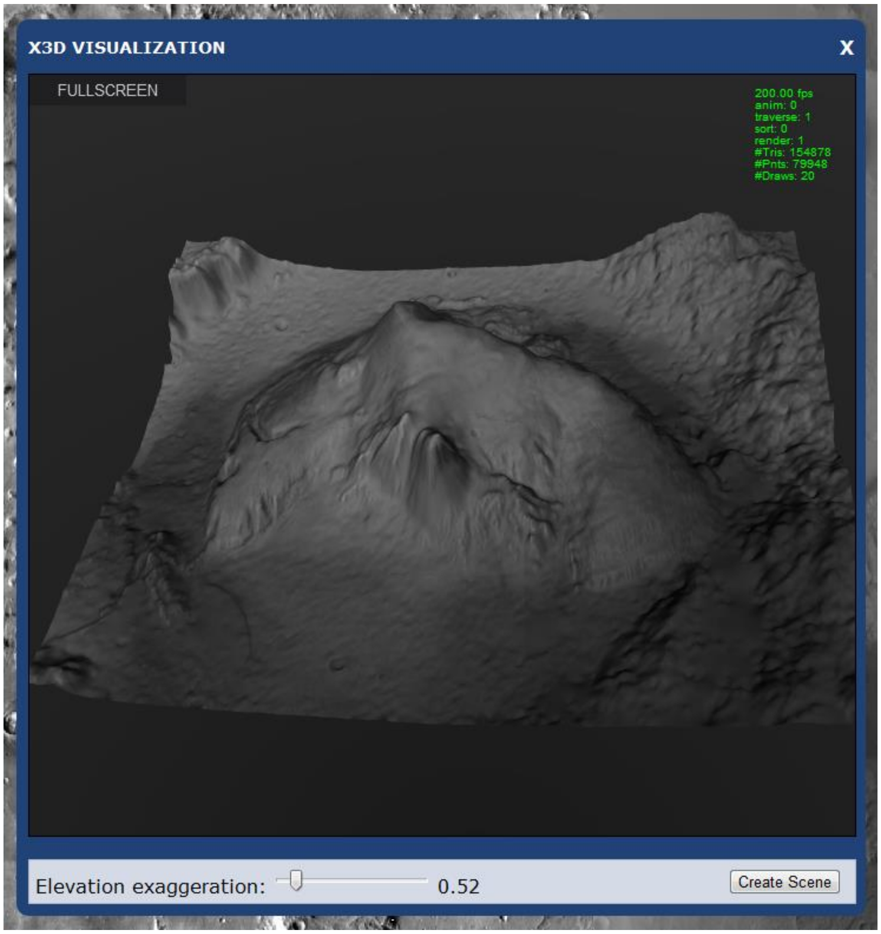
In the window toolbar the X3D and diagram windows can be shown or hidden. The background OpenLayers map shows a Gale HRSC mosaic.

X3D visualization

1. The X3D window:



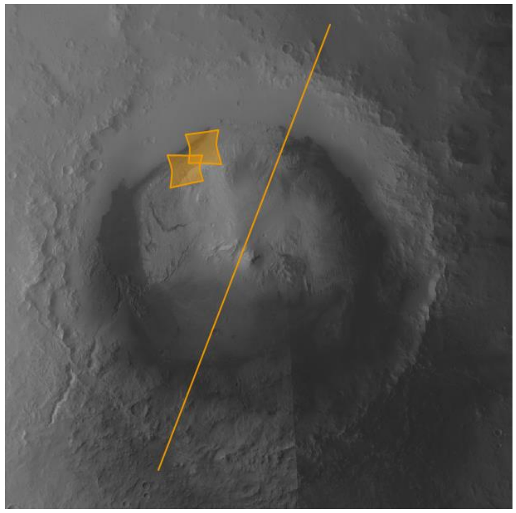
2. It includes a FULLSCREEN button, an elevation slider and a Create Scene button.
3. Click on Create Scene after which a 3D model of the mountain inside Gale Crater is loaded:



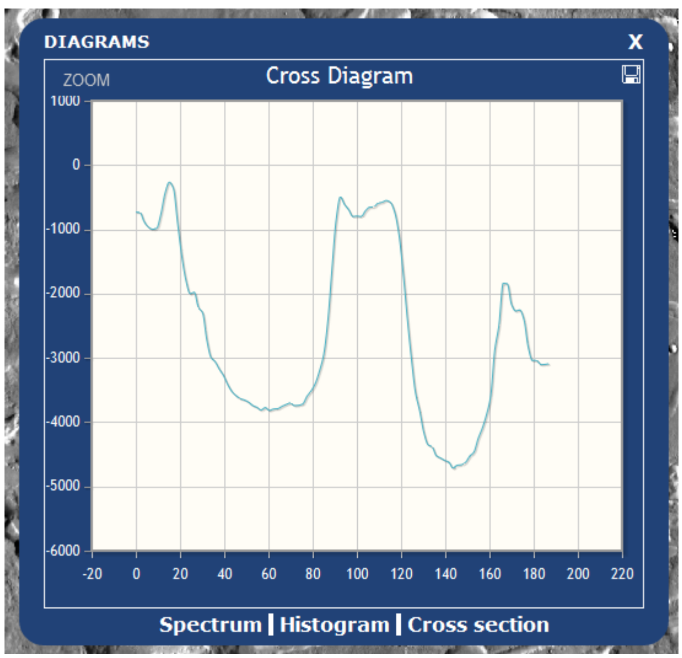
4. After loading the left mouse click allows for changing the viewpoint orientation. Double clicking moves the orientation.

Extracting Cross Section

1. Clicking on the Cross Section button ( ) in the toolbar allows two clicks within the boundary of the HRSC data of Gale Crater:



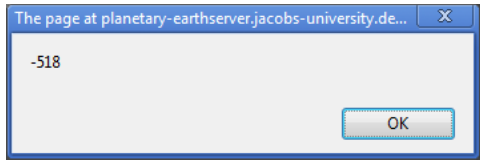
2. It will then load the respective elevation cross section in the diagram window:



3. The Y axis is in meters and the X axis in kilometers.
4. The save as button ( ) in the diagram window can be used to save the cross section to a .CSV file. This can then be loaded into software such as Microsoft Excel.

Extracting elevation value

The Z button (Z) will extract the elevation value of the location the user clicks:



CRISM data analysis

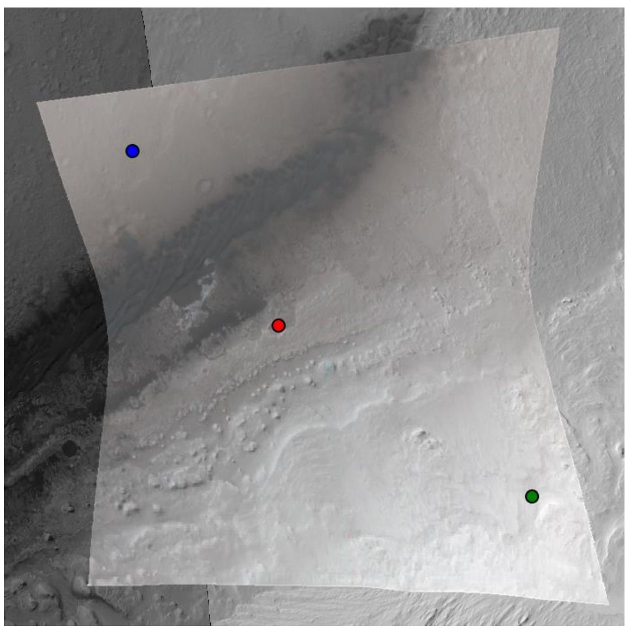
On the map please select one of the two CRISM footprints and click on Load CRISM. Now select the detector type:

1. Visual to Near InfraRed (VNIR): 107 bands
2. InfraRed (IR): 438 bands

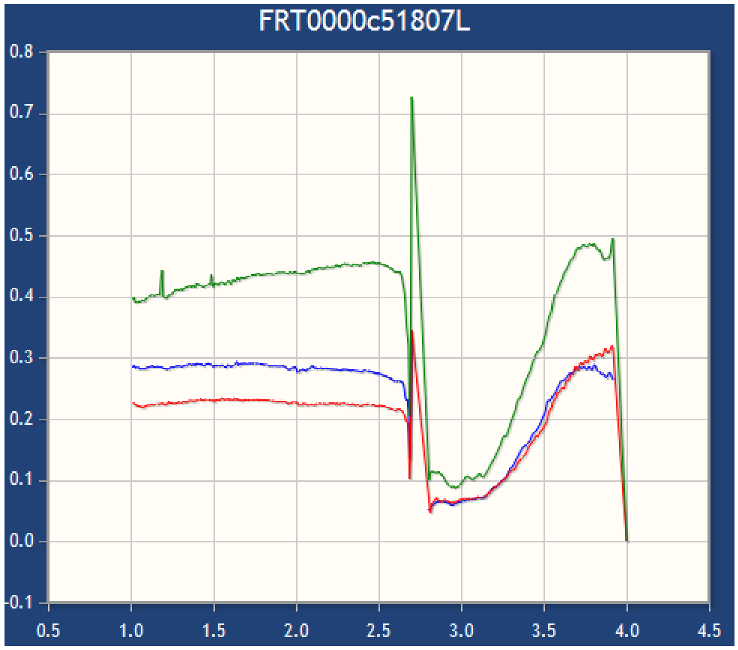
Click OK and wait until the loading is finished:

Spectrum

1. Select the Spectrum button ( ). Now click on the CRISM image in the map and a colored icon will appear where you clicked:



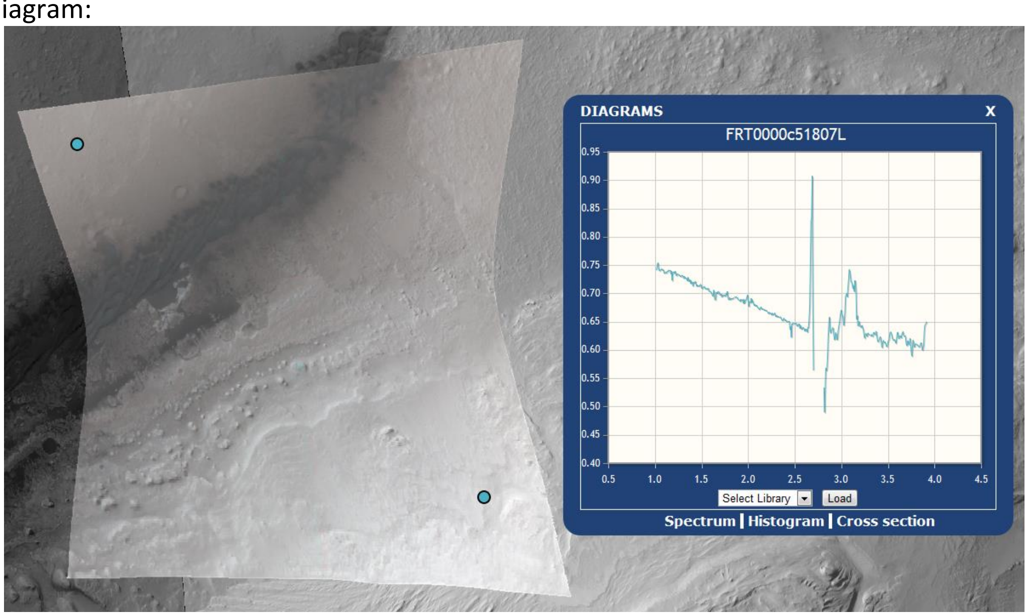
2. The spectrum of that location is added to the right diagram (with the same color):



3. You can have five different spectra with five different colors. Click the 6th time and the 1st is replaced with the 6th.
4. The save as button () in the diagram window can be used to save the spectra to a .CSV file. This can then be loaded into software such as Microsoft Excel.

Spectral ratio

1. Select the 'Spectral ratio' button (½). You can now click on two locations in the CRISM image on the map and the spectrum ratio of both locations is determined and displayed in the diagram:

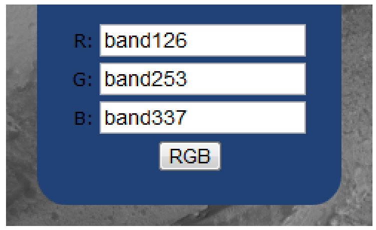


2. Click again two times to create a new ratio.

3. Also here the save as button () in the diagram window saves the spectral ratio as a .CSV file.

RGB image

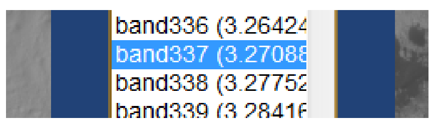
1. Click on three bands in the Bands window and see the R,G,B values change.
2. Then click on RGB.
3.



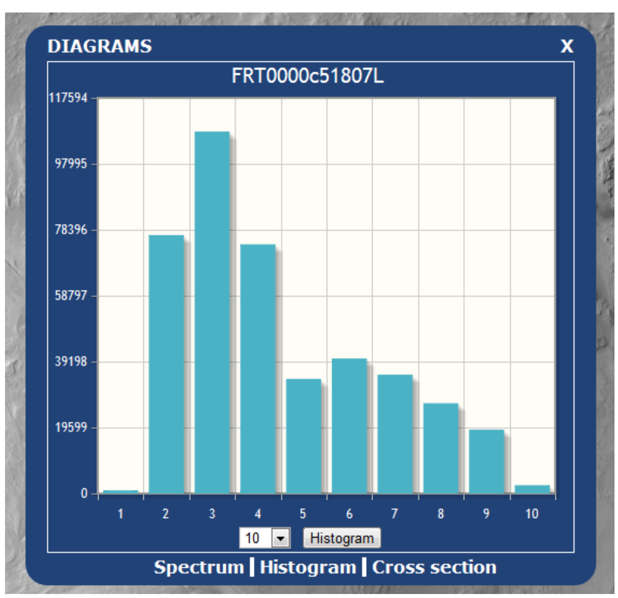
4. Please wait until the image has been loaded. It will be shown in the map.
5. The Table of Contents will now show the image name (which is the calculation made). When needed the image can be hidden.

Histogram

1. Select a band in the BANDS window:



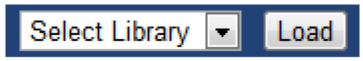
2. Go to the Histogram tab of the Diagram window, select a histogram bin size and click on Histogram. Please wait until the loading has finished.
3. After loading you will see a histogram of that band:



4. A histogram .CSV file van be saved using the save as button () in the diagram window

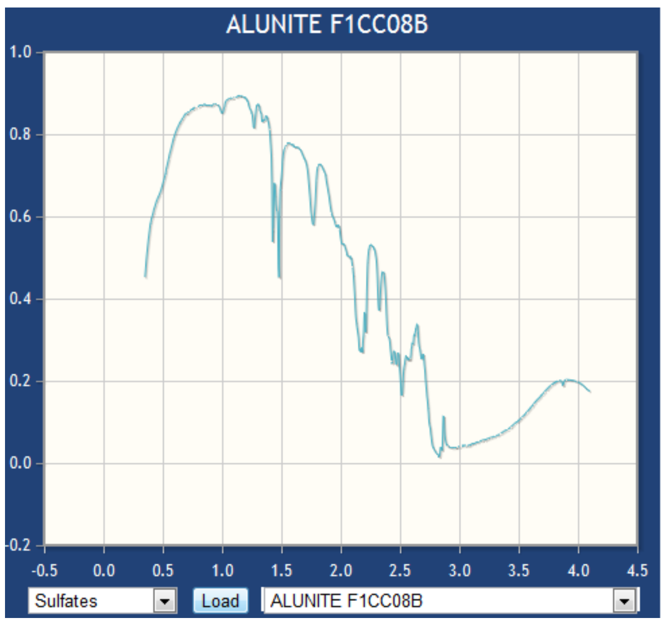
Spectral library

1. The spectrum tab in the diagram window can load spectra from a spectral library. Select a library and click on Load:

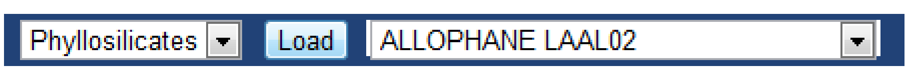


2. A second selection box appears:

3. Select the spectrum and click Load again:



4. You can select a new library and click Load again which will load the library spectra:



5. The save as button () in the diagram window can be used to save the library spectrum as a .CSV file.

Console

In the CONSOLE window you can currently do:

1. variables()

Javascript function to show all the dataset variables available.

2. p() and s()

Javascript functions to show the point coordinate (p) and spectrum (s) of the points clicked using the Spectrum button.

WCPS> s(1)
WCPS> p(1)
WCPS> p(1).lon
WCPS> p(1).lat

3. spectrum() and addspectrum()

Javascript functions to handle spectra. Spectrum() enables spectral calculations of the points clicked using the Spectrum button, or points added using addspectrum(). Within the spectral calculations javascript math functions such as Math.cos() can be used.

WCPS> spectrum(s(1))
WCPS> spectrum(s(1) / s(2))
WCPS> spectrum(Math.cos(s(1)))
WCPS> addspectrum(**LON**,**LAT**)
WCPS> addspectrum() (and, before ENTER point the mouse to the location of interest)

4. image() and rgbimage()

Javascript functions to create a greyscale and RGB image. Please wait until the image has been loaded. It will be shown in the map. The Table of Contents will now show the image name (which is the calculation made). When needed the image can be hidden.

WCPS> image("band34")
WCPS> image("band45 * band78")
WCPS> image("band74 / band24")
WCPS> image(nm2band(2300))
WCPS> rgbimage("band34","band55","band67")

5. CRISM CAT summary products

Most of the CRISM CAT (http://geo.pds.nasa.gov/missions/mro/crism.htm) summary products (Pelkey et al., 2007) have been converted from IDL to javascript strings:

WCPS> olindex
WCPS> lcpindex
WCPS> hcpindex
WCPS> bd530
WCPS> bd860
WCPS> bd920
WCPS> bd2100
WCPS> bd2200
WCPS> bd1900r
WCPS> d2300

```
WCPS> sindex
WCPS> bd1980
WCPS> doub2200
WCPS> bd2230
WCPS> bd2500
WCPS> irr3
WCPS> bd2700
WCPS> r2700
WCPS> irr2
WCPS> bd2600
WCPS> bd2350
WCPS> bd2000co2
WCPS> bd1400h2o
WCPS> bd1270o2
WCPS> cindex
WCPS> bd3400
WCPS> bd3200
WCPS> bd3100
WCPS> bd3000
WCPS> bdcarb
WCPS> icer2
WCPS> bd1900
WCPS> bd1750
WCPS> icer1
WCPS> bd1500
WCPS> bd1435
WCPS> islope1
WCPS> ira
WCPS> r770
WCPS> rbr
WCPS> sh600
WCPS> bd640
```

The strings can be used inside image() or rgbimage():

```
WCPS> image(sindex)
WCPS> rgbimage(bd1900r, d2300, sindex)
```

To derive a manual band depth you can use banddepth(low, center, high) :

```
WCPS> banddepth(800, 920, 984)
```

6. min() and max()

Javascript functions to calculate the minimum and maximum of a band or query:

```
WCPS> max(“band100”)
WCPS> min(“band100”)
WCPS> max(“band200*band100”)
```

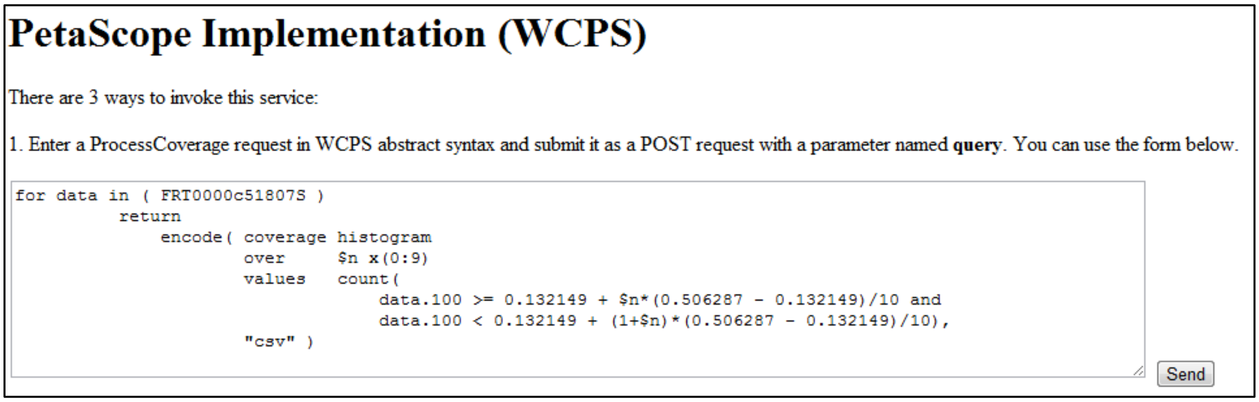
7. histogram()

Javascript functions to calculate a histogram:

```
WCPS> histogram(“band100”, 10)
WCPS> histogram(“band200”, 100)
WCPS> histogram(“band100 / band200”, 10)
```

WCPS examples

The following WCPS examples can be added to the top input box at http://planetserver.jacobs-university.de:8080/petascope/wcps. Click on Send and you immediately see the result.



CRISM Spectrum

```
for data in ( FRT0000C51807S )
    return
        encode(data[x:”URL”(137.53:137.53),y:"URL"(-4.64:-4.64)], "csv")
```

With **URL** = http://kahlua.eecs.jacobs-university.de:8080/def/crs/PS/0/1/

Gale HRSC DTM mosaic elevation value

```
for data in ( galehrscdtm )
    return
        encode(data[x:”URL”(137.53:137.53),y:"URL"(-4.64:-4.64)], "csv")
```

With **URL** = http://kahlua.eecs.jacobs-university.de:8080/def/crs/PS/0/1/

CRISM band minimum

```
for data in ( FRT0000C51807S )
    return
        encode(min(data.100), "csv")
```

CRISM band maximum

```
for data in ( FRT0000C51807S )
    return
        encode(max( (data.100 != 65535) * data.100), "csv")
```

Using max(data.100) would return 65535, the NoData value.

CRISM greyscale image

```
for data in ( FRT0000C51807S )
    return
        encode(
           (char) (255 / (max( (data.100 != 65535) * data.100) -
           min(data.100))) * (data.100 - min(data.100)),
    "png")
```

CRISM RGB image

```
for data in ( FRT0000C51807S )
    return
        encode( {
            red: (char) (255 / (max( (data.10 != 65535) * data.10) -
                 min(data.10))) * (data.10 - min(data.10));
            green: (char) (255 / (max( (data.50 != 65535) * data.50) -
                   min(data.50))) * (data.50 - min(data.50));
            blue: (char) (255 / (max( (data.100 != 65535) * data.100) -
                  min(data.100))) * (data.100 - min(data.100))},
    "png" )
```

CRISM band histogram

```
for data in ( FRT0000C51807S )
    return
         encode( coverage histogram
             over     $n x(0:9)
             values   count(
                 data.100 >= MIN + $n*(MAX - MIN)/10 and
                 data.100 < MIN + (1+$n)*(MAX - MIN)/10),
    "csv" )
```

MIN and **MAX** need to be values, min(data.100) for example cannot be used. Here is the example with values:

```
for data in ( FRT0000C51807S )
    return
         encode( coverage histogram
             over     $n x(0:9)
             values   count(
                 data.100 >= 0.132149 + $n*(0.506287 - 0.132149)/10 and
                 data.100 < 0.132149 + (1+$n)*(0.506287 - 0.132149)/10),
    "csv" )
```

References

Pelkey, S.M., Mustard, J.F., Murchie, S., Clancy, R.T., Wolff, M., Smith, M., Milliken, R., Bibring, J.-P., Gendrin, A., Poulet, F., Langevin, Y., Gondet, B., 2007. CRISM multispectral summary products: Parameterizing mineral diversity on Mars from reflectance. Journal of Geophysical Research 112.